REMARKS

Claims 1-30 remain in the application.

The rejections raised in the 11-19-2003 Office Action are respectfully traversed, and reconsideration is respectfully requested in light of the following remarks.

Section 102(e) Rejection: McCauley

Claims 1, 20, and 22 stand rejected under 35 U.S.C. §102(e) as being anticipated by McCauley (U.S. Patent No. 6,263,392).

Claim 20 is directly dependent on independent claim 1. Independent claim 1, defines a system for mapping an input device's controls with an application. The system includes a user input device having a plurality of controls. The system also includes an application that implements actions in response to activation of the controls of the user input device. The system further includes **an application program interface that receives calls from the application, the application program interface including a call that creates an association between actions in the application and the controls on the input device**, wherein creating the association comprises considering semantics related to the actions in the application.

Claim 22 defines a method of communicating between an input device and an application in a system. In an issuing action, a call is issued from the application to enumerate a suitability of input devices installed in the system, the call including an array of actions that the application uses. **In an examining action in response to the application call, the input devices installed on the system are examined by comparing controls on the input devices with actions used by the application**. In a ranking action, the input devices are ranked based on the comparison. In a providing action, the application is provided with at least the highest ranked input device that most closely matches the actions of the application.

The Office alleges that McCauley discloses an application program interface (API) that receives calls from the application, citing the following as basis for the allegation: "[the] interface controller 58 continuously monitors the USB 8 for periodic receipt of a refresh message from the host PC 4." McCauley, col. 7, lines 34-35. As understood by the Applicant, the Office is attempting to say that McCauley's interface controller 58 is like the Applicant's API and that McCauley's USB 8 of the host PC is like the Applicant's application. This, respectfully, appears to be a stretch, since *McCauley's interface controller 58 is on the input device side of the universal serial bus (USB) 8, and not on the host PC side 4.* See FIGS. 2 and 3 in McCauley. An API, as disclosed by the Applicant resides on a host side, not on the input device side. It makes no sense, even in the broadest interpretation, to say that McCauley's interface controller 58 is an API. It also makes no sense to say that the hosts' USB 8 is the application. In fact, in the second paragraph on page 4 of the Office Action, the Office seems to understand this point, when saying that the application is a game application. It is not logical for the Office to inconsistently define elements like this.

McCauley does not teach or suggest **an application program interface that receives calls from the application, the application program interface including a call that creates an association between actions in the application and the controls on the input device** as claimed by the Applicant as part of independent claim 1. McCauley's interface controller 58, even if it were incorrectly interpreted to be an API, certainly does not receive calls from an application as defined by the Applicant. While the interface controller 58 may receive a refresh message from the host PC's USB Port 24, this could hardly be construed to be a call from an application running on the PC for execution of a remote procedure. McCauley's refresh message is simply a watchdog-related signal which is built into the USB standard, and again emphasizes that the interface controller 58 in McCauley is not an API, but an external, peripheral hub-like USB device which accepts not only USB inputs but also analog inputs in order to digitize and serialize analog data and provide it to a host PC over the USB. McCauley simply discloses a way to connect the external input, or control information to a PC. McCauley in no way provides for creating an association between actions in an application on a host device and the controls on an input device.

McCauley does not teach or suggest **an examining action where in response to the application call, the input devices installed on the system are examined by comparing controls on the input devices with actions used by the application** as claimed by the Applicant as part of independent claim 22. As discussed above, the Office erroneously suggests that a refresh message from McCauley's host PC to the interface controller 58 is the same as the Applicant's application call. Even if we were to incorrectly accept this assumption, McCauley falls short in at least two other respects. First, McCauley lacks any teaching or suggestion of an examination of the input devices by comparing controls on the input devices with actions used by any application. McCauley is not concerned with applications running on the system, since it simply provides a method to provide serial input to a host PC. McCauley is silent as to what the host PC will do with this data or how an application on the host PC might deal with the serial data. Second, the incorrect assumption that a refresh message from the USB (acting as the application according to the Office) is a call from an application makes no sense in light of part of the Applicant's claim where controls on the input device are compared with actions used by the application. The USB does not care what control actions are being communicated; the USB simply serializes data and passes it from the input device to the host PC. McCauley does not teach or suggest each and every element of any of the Applicant's claims.

While the foregoing is sufficient to demonstrate that McCauley does not teach or suggest the matter claimed in the current application, the Applicant also wishes to dispute the other assertions of the Office with respect to McCauley. In particular, the allegations in the Office Action that McCauley discloses at column 3, lines 1-21 an action for creating an association between actions in an application and the controls on an input device by considering semantics related to the actions in the application is incorrect. "A preferred embodiment of the method of the present invention includes the reading of peripheral state information of certain peripheral devices by an interface control module, where the peripheral state information may be indicative of an instantaneous mechanical position of an analog potentiometer and/or indicative of an alternately manually depressed and released button. The interface control module transmits formatted data packets, such as HID report descriptors, and formatted signal data, such as HID reports, to the host computer via a communications interface, or bus. The HID report descriptors are substantively formatted according to the HID standard and the archetype and/or structure of the corresponding peripheral device. Each HID report is formatted substantively in accordance with the HID standard and a reading of the state information as read from a specific peripheral device.

A system software of the host computer interprets the significance of the formatted data packets and the formatted signal data according to a pre-established communication standard, such as the HID standard, or another suitable knowledge based data communications formatting standard or technique known in the art." McCauley, col. 3, lines 1-21.

The foregoing citation actually makes the argument against McCauley. In McCauley, data from the input devices is formatted according to a pre-established communication standard, such as HID for devices connected to the host PC over the USB. McCauley does not describe interpretation of the HID data by applications, and does not really need to, since McCauley is concerned with a hub-like device for serializing a variety of input device control data onto a USB line for connection to a host PC.

Furthermore, there is absolutely no mention in McCauley of semantics. The Office alleges that the "peripheral state information" referred to in McCauley at col. 3, lines 1-50 is equivalent to the semantics related to the actions of the application as claimed by the Applicant. There is absolutely no basis to support this interpretation. The Office is insisting that *peripheral* state information is the same as semantics related to the actions of the *application* on the host PC. The Applicant respectfully points out that the peripheral is a different element from the application on the host PC. Respectfully, the Office appears to have inadvertantly twisted terms to fit the claims of the current application, without regard as to whether the terms pertain to the host PC or whether they pertain to some external device, such as an input device like McCauley's analog-compatible hub.

The "Response to Arguments" section of the Office Action (section 16) says that operation of the HID produces a data structure with a report descriptor having a label that expresses a behavior that an application exhibits, corresponding to an instantaneous mechanical position of an input device element, whereby each element of the data structure contains a description of a control, equivalent to said semantic." Office Action, page 15, lines 1-6. This is not correct. These HID labels for the input devices pertain to actions of the input device, not an application. The Applicant respectfully challenges this assertion of knowledge by the examiner and requests that the Examiner provide an affidavit in support of such well-known prior art. See MPEP 2144.03. This should facilitate prosecution in the event that an appeal should become necessary.

Similarly, the Office states without support or reference to McCauley that "McCauley teaches of a library of predefined peripheral device archetypes, equivalent to said Genres, wherein an interpretive software system interprets the semantic or peripheral state information produced by the input device and sends it to the application as HID report descriptors of a specific genre for application processing." Office Action, page 4, 1st paragraph. The Applicant contests the Examiner's statement that the HID peripheral archetypes are equivalent to "genres" as used in the current application. "A genre is a semantic vocabulary that encapsulates the common input elements among applications falling into a broad category." Specification, page 11, lines 7-9. Example genres, as described in the specification, might be a driving game genre or a role playing genre. Clearly a genre is related to types of applications which run on a host. In contrast to this, the Office appears to say that genres are categories of peripheral devices. This is incorrect and completely at odds with the teachings and claims of the current application.

Therefore, the Applicant believes independent claims 1 and 22 are allowable over McCauley, and respectfully requests that the 35 U.S.C. 102(e) rejection based on McCauley be withdrawn. Similarly, dependent claim 20 is believed to be allowable over McCauley based on the allowability of its base claim and for the features set forth therein.

Section 102(e) Rejection: Chan

Claims 1-19 and 21-30 stand rejected under 35 U.S.C. §102(e) as being anticipated by Chan et al. (U.S. Patent No. 5,991,546, hereafter referred to as 'Chan').

Claims 2-19 and 21 are directly or indirectly dependent on independent claim 1, the features of which were discussed above with respect to McCauley.

Claims 23-26 are directly or indirectly dependent on independent claim 22, the features of which were discussed above with respect to McCauley.

Claims 28-29 are directly or indirectly dependent on independent claim 27. Independent claim 27 defines a method for mapping an input device's controls with an application in a system. In response to a request from an application program to create an action-to-control mapping, in a reading action, stored user preferences for the action-to-control mapping are read and a stored default file that includes manufacture provided defaults for the action-to-control mapping is read. In another reading action, a structure that includes action values and action semantics associated with the action values is read, the action values being defined by the application. **In a using action, the stored user preferences and the stored default file are used to create an association between the action values associated with the application and the controls on the input device**.

Independent claim 30 defines a computer-readable medium including computer-executable instructions to perform a method for using a computer input device with a software application, including an application program interface, responsive to a call from an application, that returns an enumeration of input devices that substantially match the actions of the application; and **an application program interface, responsive to a call from the application, that uses one of the enumerated input devices selected by the application to build an action-to-control mapping**.

The Office's characterization of Chan is incorrect. Chan details a processing capability 6 which accepts connections from multiple types of input peripherals 7, 11, 15, 19, 23, 27, 10, 12, 24 (see Chan, FIG. 1) serializes the input data, and then provides the input data over a USB 14 to a host 16. The Office alleges that this processing capability 6 is the API (Office Action, page 6, sec. 7, line 7) and that the host 16 is the application (Office Action, page 6, sec 7, lines 3-5).

The Office also alleges that Chan teaches or suggests the API includes a call that creates an association between actions in the application and the controls on the input device. (Office Action, page 6, sec 7, lines 7-11). Using the Office's definitions of application as the host 16 and the API as the processing capability 6, this would mean that Chan's processing capability 6 includes a call that creates an association between the actions of the host 16 and the controls of the input device. No reading of Chan can support this. The processing capability 6 does not create an association of actions of the host 16 and the controls of the input device. The processing capability 6 manages data flow. For example, if the host 16 requests data from the mouse endpoint, then the processing capability 6 simply responds with any mouse data available. See Chan, col. 5, lines 35-50. This processing capability 6 is not creating associations between actions in the application and the controls on the input device. Chan does not teach or suggest that the processing capability ever receives any information about the applications running on the host. The processing capability 6 is merely a hub meant to provide data to the host from various input devices.

As such, and at the very least, Chan does not teach or suggest an application program interface that receives calls from the application, the application program interface **including a call that creates an association between actions in the application and the controls on the input device** as claimed by the Applicant as part of independent claim 1.

Similarly, Chan does not teach or suggest an examining action where in response to the application call, **the input devices installed on the system are examined by comparing controls on the input devices with actions used by the application** as claimed by the Applicant as part of independent claim 22.

The Office appears to go through a detailed analysis of how Chan teaches or suggests the matter of dependent claims 2-21 and 23-26. While the Applicant appreciates the efforts of the examiner, these detailed discussions are flawed from the start based on the incorrect interpretation of Chan as discussed above with regard to independent claims 1 and 22.

The Office provides further incorrect analysis with regard to Chan. The Office alleges that Chan creates an action to control mapping in response to a request from the application. See Office Action page 11, sec. 11, lines 2-3. As explained above, Chan's host generates a request for data from the processing capability 6. The returned data is input data, for example, what key or combination of keys has been pressed, as generated by an input control device. The data processing capability 6 does not know anything about the actions which an application on the host might take with the control data.

The Office's citations in Chan (col. 5, lines 14-50; col. 6, lines 30-45) are offered to incorrectly support an assertion that Chan teaches or suggests any type of action-to-control mapping. According to the current case, actions are under the direction of the application, while controls are available on the input device. The areas of Chan referred-to by the Office as proof of action-to-control mapping are simply incorrect. In one instance, Chan describes a data adaptation unit, such as serial EEPROM 24. Chan col. 5, lines 14-50. The data adaptation unit 24 does not have any knowledge of actions which are under direction of the applications on the host. As such, it should be clear that this data adaptation unit 24 can not perform any type of action to control mapping. Additionally, the Office is incorrect when it says Chan's EEPROM 24 contains semantic mappings. Chan does not contain any reference to the term "semantic" or "semantics". A semantic is a label that expresses a behavior that an application exhibits upon operation of a control. Spec, page 11, lines 9-10. It is ridiculous to say that Chan's EEPROM contains mappings for labels that express behaviors of an application upon operation of a control. The EEPROM does not know anything about the application and could not performing any mapping or association related to the application.

At a minimum, Chan does not teach or suggest **a using action where the stored user preferences and the stored default file are used to create an association between the action values associated with the application and the controls on the input device** as claimed by the Applicant as part of independent claim 27.

Similarly, Chan does not teach or suggest **an application program interface, responsive to a call from the application, that uses one of the enumerated input devices selected by the application to build an action-to-control mapping** as claimed by the Applicant as part of independent claim 30.

Therefore, the Applicant believes independent claims 1, 22, 27, and 30 are allowable over Chan, and respectfully requests that the 35 U.S.C. 102(e) rejection based on Chan be withdrawn. Similarly, dependent claims 2-19, 21, 23-26, and 28-29 are believed to be allowable over Chan based on the allowability of their base claims and for the features and method actions set forth therein.

Section 103(a) Rejection: McCauley and Chan

Claim 27 stands rejected under 35 U.S.C. §103(a) as being unpatentable over McCauley in view of Chan.

The features of independent claim 27 have been discussed above with regard to Chan. McCauley does not overcome the shortcomings of Chan. In addition to the above discussion of McCauley, at a minimum, the Office incorrectly alleges that McCauley teaches reading a structure that includes action values and action semantics associated with the action values, the action values being defined by the application. See Office Action, page 13, sec. 15, lines 4-6. In support of this position, the Office cites McCauley col. 4, lines 1-5 and col. 3, lines 28-50. These citations describe the formation of HID reports, organized by pre-defined HID class or peripheral types in which peripheral data is sent to the host. No reference is made to an application on the host, only the peripherals. McCauley does not teach or suggest action semantics. As such, it is clear that McCauley, like Chan, does not teach or suggest **a using action where the stored user preferences and the stored default file are used to create an association between the action values associated with the application and the controls on the input device** as claimed by the Applicant as part of independent claim 27. Therefore, the Applicant believes independent claim 27 is allowable over McCauley and Chan, whether taken individually or in combination.

Challenge to the non-supported assertions in the Office Action

Throughout the Office Action, the Examiner alleges that the HID interface referred-to in McCauley issues calls that include an array of actions used by applications, and in response to such a call, examines input devices installed on the system by comparing controls on the input devices with actions used by applications. The Applicant believes that such statements are untrue, and illogical, however, the Office Action repeatedly contains such statements and refers the Applicant to sections of McCauley which broadly refer-to the HID. See, for example, the Office Action, last paragraph of page 4 and the text on the top five lines of page 5. Since McCauley does not actually contain such interpretations of the HID standard, it appears that the Examiner is trying to state that these interpretations of the HID standard are well-known in the prior art based on knowledge possessed by the Examiner. As such, the Applicant respectfully challenges this statement and requests that the Examiner provide an affidavit in support of such well-known prior art. See MPEP 2144.03.

Conclusion

Therefore, claims 1-30 are believed to be in a condition for allowance, and an early action to this end is respectfully solicited. If the examiner should have any questions, a call to the undersigned registered patent agent is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

By Christopher B. Miller
Christopher B. Miller
Registration No. 48,398

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 226-7391
Facsimile: (503) 228-9446